ENGINEERING CHANGE REQUEST FORM

Requester Details

Requested By:	Ramya N	ECR Number:	ECR/12/9/2018---8
Employee Code:	QA23	Site:	Corporate
Job Position	Manufacturing Engineer	Entity:	Corporate

Change Details

Change Title:	Change in 4299 Project Test in engineering Change Request (Design Change).		
Project Name:	Engine Delta hawk EP503-EP	Part Name:	Front Impact Airbag System
Requested Date:	9/12/2018 3:53:23	Proposed Closure Request Date:	9/30/2018

Change Description

Change Category:

- [x] Design Engineering
- [x] Document Control
- [x] Manufacturing Engineering
- [ ] Master Schedulers
- [ ] Material Planners
- [ ] Production Schedulers
- [ ] Program Managers
- [ ] Quality Engineering

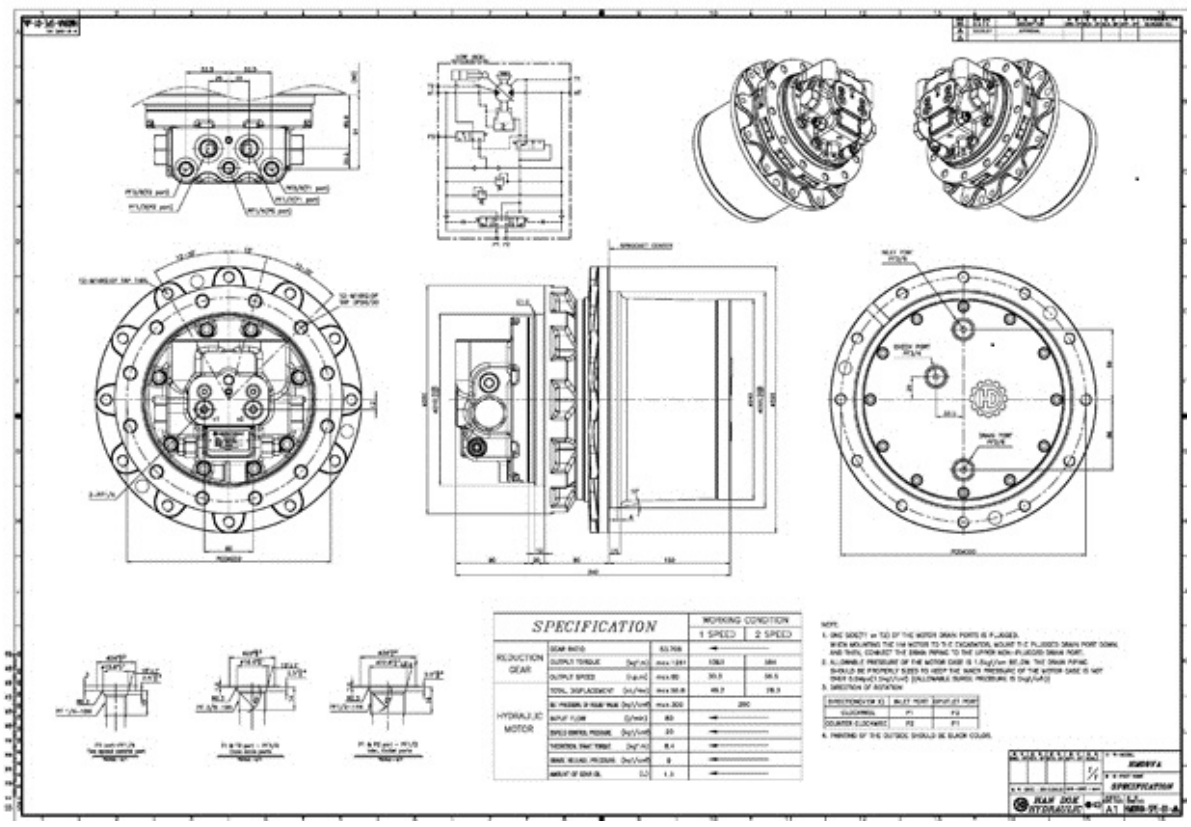
Type of Change:

- [x] Major Change (Class A) (Production line cannot run without sign off)
- [ ] Major Change (Class B) (Production line can run after QC Associate has verified the parts are within specification. RFC form can then be circulated for approval.)
- [x] Minor Change (Class C) (Production line can run if dimensions are within specification. RFC form can be submitted with the first off part to QC.)

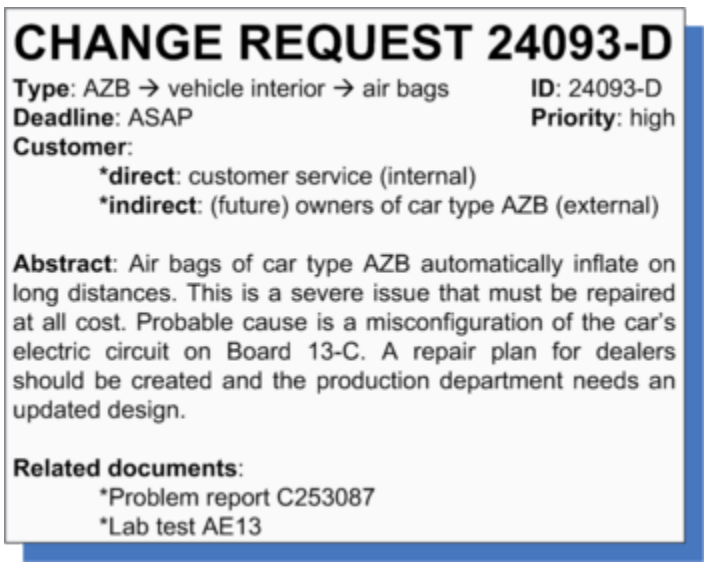
Reason:

Major Change (Class A) (Production Line cannot run without Sign Off Major Class b
Major Change (Class B)
Major Change (Class C) (Production Line can if dimensions are with in specification . RFC Form can be submiited with the first off Part to QC.

Area of change:



A good example of the change management process in action can be found inÂ software development. Often users report bugs or desire new functionality from their software programs, which leads to aÂ change request. TheÂ product softwareÂ company then looks into the technical and economical feasibility of implementing this change and consequently it decides whether the change will actually be realized. If that indeed is the case, the change has to be planned, for example through the usage ofÂ function points. The actual execution of the change leads to the creation and/or alteration ofÂ software codeÂ and when this change is propagated it probably causes other code fragments to change as well. After the initial test results seem satisfactory, the documentation can be brought up to date and be released, together with the software. Finally, the project manager verifies the change and closes this entry in the change log.



Another typical area for change management in the way it is treated here, is theÂ manufacturingÂ domain. Take for instance the design and production of aÂ car. If for example the vehicleâ€™s air bags are found to automatically fill with air after driving long distances, this will without a doubt lead to customer complaints (or hopefully problem reports during the testing phase). In turn, these produce a change request (see Figure 2 on the right), which will probably justify a change. Nevertheless, a â€“ most likely simplistic â€“ cost and benefit analysis has to be done, after which the change request can be approved. Following an analysis of the impact on the car design and production schedules, the planning for the implementation of the change can be created. According to this planning, the change can actually be realized, after which the new version of the car is hopefully thoroughly tested before it is released to the public.

Benefits of Change:

- ☐ Capacity
- ☐ Customer Satisfaction
- ☐ Improve productivity
- ☐ Improve quality
- ☐ Prevent defect
- ☐ Reduce unit cost
- ☑ Technical Development

Risk Analysis

Select Risk Template : Risk and Opportunity — View Risk

Dimensions (if Required)

- ☐ Design Engineering
- ☑ Document Control
- ☐ Manufacturing Engineering
- ☑ Master Scheduler
- ☐ Materials Planner
- ☐ Program Manager
- ☐ Quality Engineering

Documents to Change

- ☑ Alerts, Deviations â€“ copy entire team, including Team Leadsonce approved.
- ☑ Approved Supplier List
- ☑ BAAN Update
- ☑ Cal Val Log
- ☑ Calibration Instruction
- ☑ Complete Validation Plan
- ☑ Cost Analysis
- ☑ Customer Drawing
- ☑ Customer or Supplier Pkg
- ☑ DFMEA Updates
- ☑ PPAP package organized and link sent to Doc Control
- ☑ Update DVPR

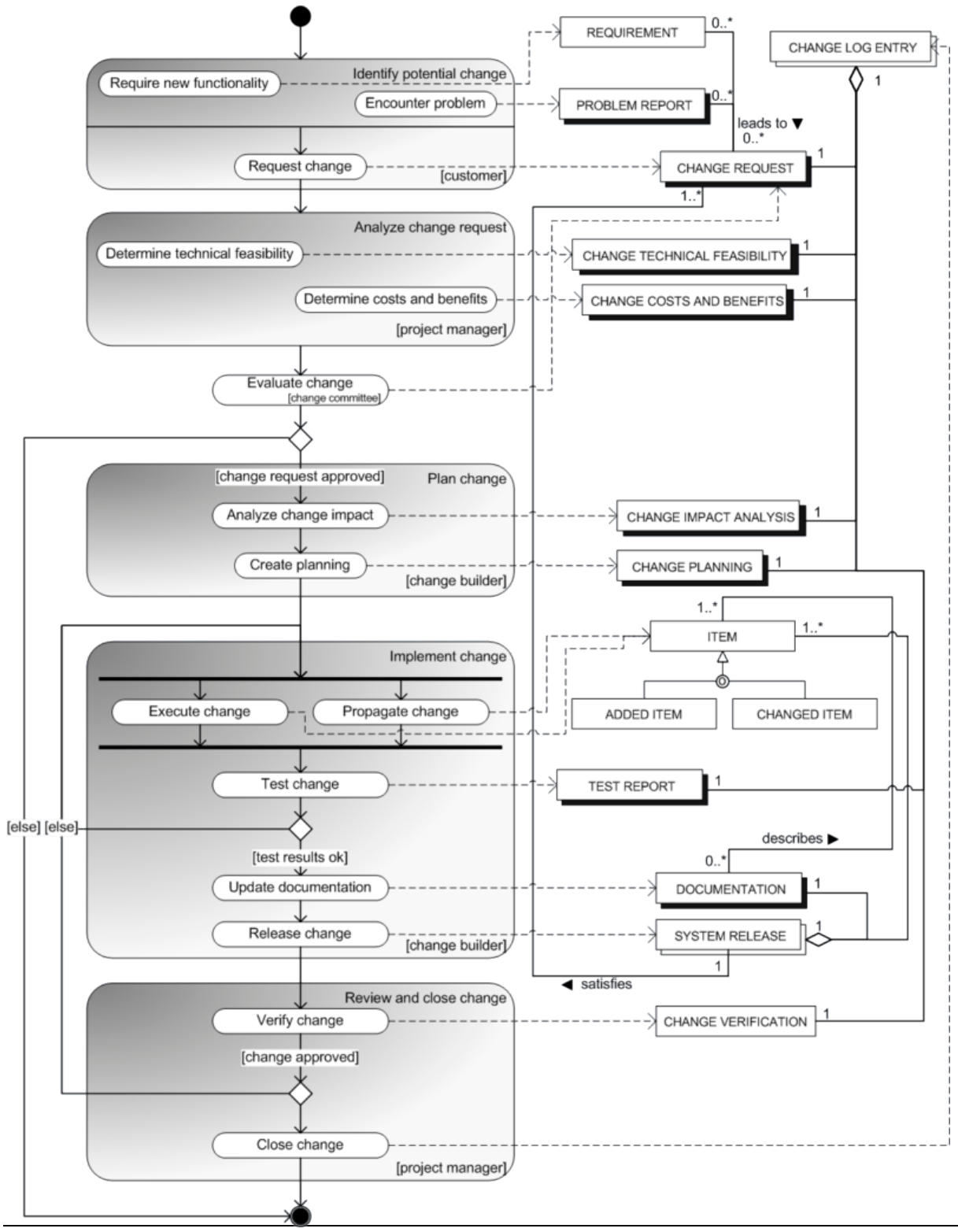
Customer Notification Required

Does this change request require customer notification? ☑

If YES, did customer approved the change requested? ☑ YES ☐ NO

Attachment: AquaProExport (4).xlsx

Remarks:



Approval Section				
Approvers	Comments	Approval Status	Rejection Status	Approved/Rejected Date
		☐	☐	

Implementation Section			
Does this change needs a new project ?	☐		
Implemented By		Implementation Date	

COMMENTS

Ramya N commented as Change management (ITSM) ... The objective of change management in this context is to ensure that standardized methods and procedures are used for efficient and prompt handling of all changes to control IT infrastructure, in order to minimize the number and impact of any related incidents upon service.
Ramya N commented as Change management is an umbrella term that covers all types of processes implemented to prepare and support organizational change. ... Change management in the context of project management often refers to a change control process when working on a project
Ramya N commented as Document that describes the requested change and why it is important; can originate from PROBLEM REPORTS, system enhancements, other projects, changes in underlying systems and senior management, here summarized as REQUIREMENTS (Dennis, et al., 2002). Important attribute: â€˜go/no-go decisionâ€™, i.e. is the change going to be executed or not?

RISK FORM

Risk and Opportunity

Objective	Failure in Meeting	Effect of Failure	Severity	Cause	Occurrence	Risk = Sev x Occ	Recommended Action	Who	Promised Date
There is considerable overlap and confusion between change management, change control and configuration management. The definition below does not yet integrate these areas. Change management has been embraced for its ability to deliver benefits by improving	The project manager determines the costs and benefits of the proposed CHANGE REQUEST, resulting in CHANGE COSTS AND BENEFITS. This and the above sub-activity can be done in any order and they are independent of each other, hence the modeling as unordered activities.	Based on the CHANGE REQUEST, its CHANGE TECHNICAL FEASIBILITY and CHANGE COSTS AND BENEFITS, the change committee makes the go/no-go decision. This is modeled as a separate activity because it is an important process step and has another role performing it. It is modeled as a sub-activity (without any activity containing it) as recommended by Remko Helms (personal communication).	40	The extent of the change (i.e. what other items the change effects) is determined in a CHANGE IMPACT ANALYSIS. It could be argued that this activity leads to another go/no-go decision, or that it even forms a part of the Analyze change request activity. It is modeled here as a planning task for the change builder because of its relationship with the activity Propagate change.	60	24	The changes resulting from Execute change have to be propagated to other system parts that are influenced by it. Because this and the above sub-activity are highly dependent on each other, they have been modeled as concurrent activities.	Mazar Khan	9/30/2018

the affected system and thereby satisfying "customer needs," but has also been criticized for its potential to confuse and needlessly complicate change administration. In some cases, notably in the Information Technology domain, more funds and work are put into system maintenance (and change management) than into the initial creation of a system.[2] Typical investment by organizations during initial implementation of large ERP systems is 15 to 20 percent of overall budget.									

COMMENTS

Ramya N commented as Besides activities, the process-data diagram (Figure 1) also shows the deliverables of each activity, i.e. the data. These deliverables or concepts are described in Table 3; in this context, the most important concepts are: CHANGE REQUEST and CHANGE LOG ENTRY. A few concepts are defined by the author (i.e. lack a reference), because either no (good) definitions could be found, or they are the obvious result of an activity. These concepts are marked with an asterisk (â€˜*â€™). Properties of concepts have been left out of the model, because most of them are trivial and the diagram could otherwise quickly become too complex. Furthermore, some concepts (e.g. CHANGE REQUEST, SYSTEM RELEASE) lend themselves for the versioning approach as proposed by Weerd,[6] but this has also been left out due to diagram complexity constraints.